Part I: Technical Roadmap

This document describes the final tasks for completing the technical part of your term project. This is in addition to Part II, which describes the project report (coming soon). Building on skills and experience from your past three group assignments, the tasks listed here expand the problem scope for unsupervised intrusion detection using time series analysis and forecasting applied to stream data from a supervisory control system. There are several goals: enriching the *feature engineering phase*, allowing for models with arbitrary many states, and advancing anomaly detection by analyzing several datasets with injected anomalies. The data analysis, the design and selection of models, and the presentation of the experimental results form the technical basis of five main aspects to be addressed in your project report (as will be detailed in the report description, i.e. Part II).

All groups will work with the same datasets for model training, testing and anomaly detection. This way, the results of your experiments will be comparable to the results of other groups and allow for ranking the achieved model performance across all groups.

Please use only the datasets listed under “Term Project” **on the course page.**

Complete the following tasks:

1. **Feature Engineering.** Choose a subset of dependent variables for training multivariate Hidden Markov models on normal electricity consumption data. For deciding on the subset of dependent variables that are most suitable for training your models, you need to perform a Principal Component Analysis (PCA)[1], a superior alternative to using a correlation matrix. Provide a proper rational for your final choice of response variables based on the result of the PCA, which is explained in more detail on the next page. Note that you need to scale the raw data using *standardization* prior to applying PCA.

2. **HMM Training and Testing.** Partition your scaled data into train and test. Choose a weekday or a weekend day and a time window between 2 to 6 hours on that day. For this time window, train various multivariate Hidden Markov Models on the train data with different numbers of states. Decide on the most suitable dependent variables for training these HMMs using your PCA results. For models with not less than 4 and not more than 20 states compare the results of log-likelihood and BIC and select the most reasonable models with an overall good fit on the train data. Note that you do not need to train a model for each and every number of states across the given range. Finally, calculate the log-likelihood of

[1] Principal component analysis is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called *principal components*.

the test data for your selected models to decide on the best one among these candidates. Note that you need to compare normalized log-likelihood of the train data and the test data. *Hint:* For calculating the log-likelihood of the test data, look at the `fit`-section on Page 15 and the `forwardbackward`-section on Page 21.

https://cran.r-project.org/web/packages/depmixS4/depmixS4.pdf

3. **Anomaly Detection.** Using the above multivariate HMM, compute the log-likelihood for the respective observation sequences associated with the same time window in each of three datasets with injected anomalies provided on the course page under Term Project. That is, for each dataset compute the log-likelihood over all instances of the time window over one full year. Compare and interpret the three datasets regarding the degree of anomalies present in each of the datasets in some detail. compare normalize
比较并详细解释关于每个数据集中存在的异常程度.

PRINCIPAL COMPONENT ANALYSIS 主分量分析

Principal Component Analysis is a useful technique for analysis of datasets with many variables. It basically is a type of linear transformation which takes a dataset with many variables (i.e., number of responses and number of samples), and simplifies it by turning it into a smaller number of variables, called *principal components*.

This technique also allows you to visualize how the data is spread out in a dataset. The underlying mathematics is somewhat complex though, so we won't go into too much detail, but PCA gives us a number (percentage) for each variable which indicates how much variance there is in the data for that variable.

To have a better understanding, let's assume you have a year worth of multivariate data which has 7 responses. Further assume you chose a time window from <start time> to <end time> on <weekday>. Therefore, you would have 52 samples for each of these 7 responses. After applying PCA on this data, you obtain 7 principal components. Each of these PCs is represented by a number which explains a percentage of the total variation in the dataset. If PC1 is 65%, it means it has 65% of the total variance; in other words, nearly two-thirds of the information in the dataset (7 variables) can be encapsulated by this one principal component.

In this part, you should (I) compute the principal components of the original dataset; (II) plot the results (PCs); (III) interpret the results. In order to compute the principal components, we recommend to use the stats package (the important commands you may need are `prcomp()` and `summary()`). To plot the result we recommend to use the **ggbiplot** package (it is based on the **ggplot** package).

Please read about **Principal Component Analysis** to gain a better understanding of this concept and also use the documentation of the packages you use. There are also three short videos on YouTube explaining PCA in a generally understandable way. You can find these videos at:

https://youtu.be/HMOI_lkzW08
https://youtu.be/FgakZw6K1QQ
https://youtu.be/0Jp4gsfOLMs

SUBMISSION

Please submit the R code of your solutions for all three tasks as well as a PDF copy of your term project report presenting and explaining your experiments and the results through the course page by 12 APRIL 2021, 23:59 PST.

30% test 70% trainning

20% test 80% trainning